

Félix Clément
Nicolas Thibault

Groupe :00

Traitement d'image
SIF1033

Phase 1
PROJET DE SESSION

Travail présenté à
François Meunier

Département de mathématiques et d'informatique
Université du Québec à Trois-Rivières
9 février 2022

Faire fonctionner le programme

Cette partie du projet a été fait avec le langage de programmation *Python*. Le script ainsi que le fichier solution du projet (*.sln* de *Visual Studio*) sont situés dans le dossier :

Tp1_Sif_1033_Félix_Clément. Voici les importations nécessaires pour faire fonctionner le script :

Pour *OpenCV* (*cv2*) :

```
pip install opencv-python
```

Pour *Matplotlib* :

```
pip install matplotlib
```

Le module *Tkinter* peut seulement s'installer à l'aide de l'installateur de *Python*.

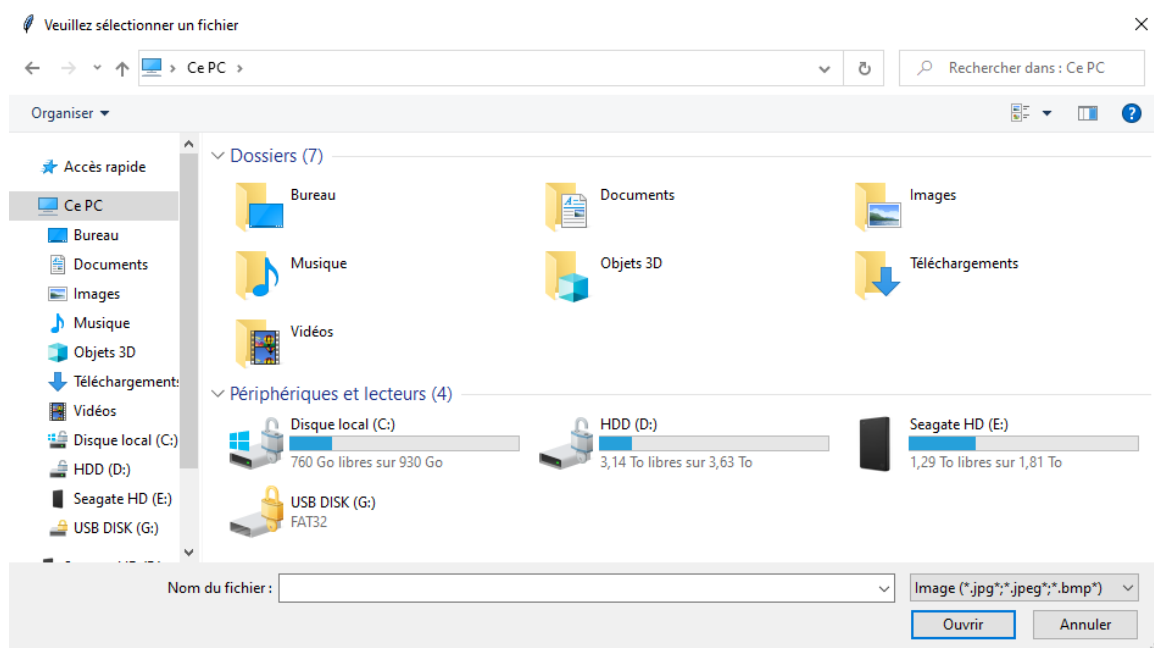
Tkinter est utilisé pour les fenêtres de dialogues et pour avoir une fenêtre pour parcourir le disque à la recherche d'un fichier. *Matplotlib* est utilisé pour lire une image sur le disque, car la fonction de *cv2* donne une erreur si la « path » du fichier est trop longue (voir : <https://stackoverflow.com/questions/68716321/how-to-use-absolute-path-in-cv2-imread>). Elle est également utilisé pour sauvegarder une image sur le disque, car la fonction *imwrite()* de *cv2* écrivait une image qui faisait 0 bytes, et elle était donc illisible.

Déroulement du programme

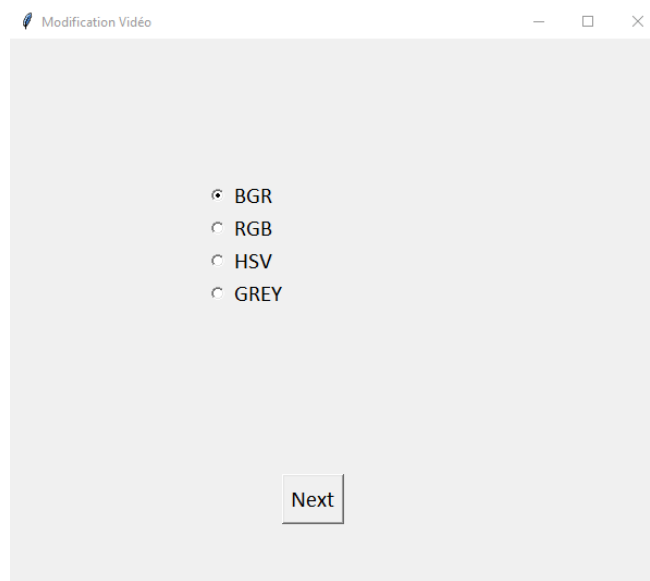
Le programme démarre avec une fenêtre permettant de choisir si on veut ouvrir une image ou une vidéo.



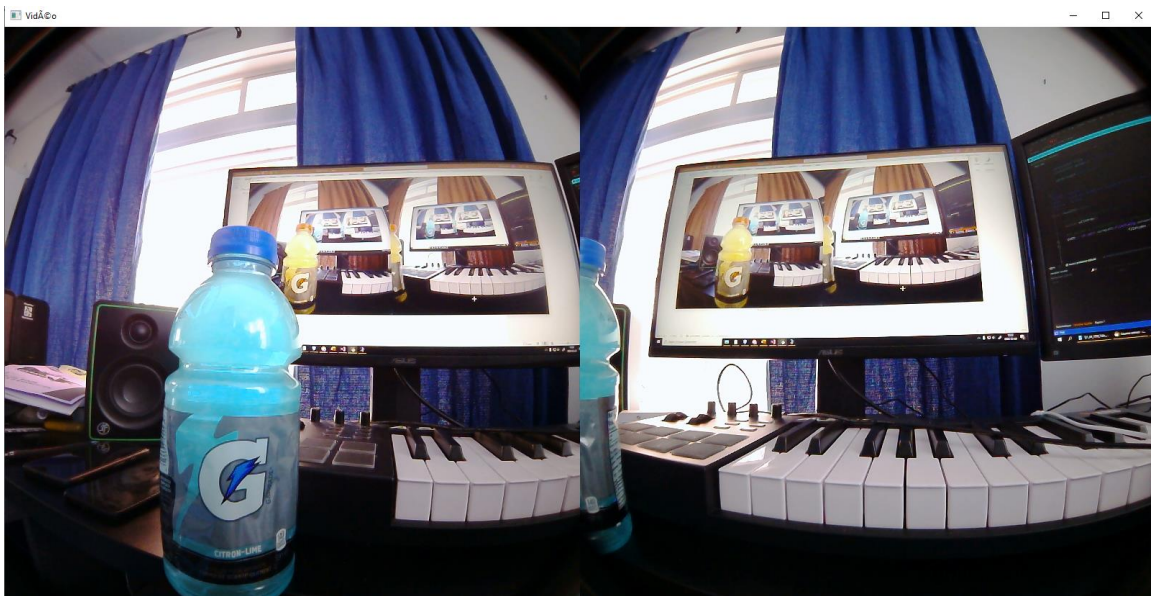
Les deux boutons font afficher une fenêtre permettant de choisir le fichier sur le disque.
Note : L'option vidéo offre la possibilité d'utiliser la webcam au lieu de choisir une vidéo. Également, certains types de fichier ont été inclus dans les choix visibles, mais on peut sélectionner une option pour voir tous les fichiers.



Si l'utilisateur à sélectionner une vidéo dans le menu, le programme va offrir d'appliquer des modifications de couleur sur la vidéo. Par exemple : en mode GBR (par défaut), en mode RGB, en HSV ou en GrayScale.

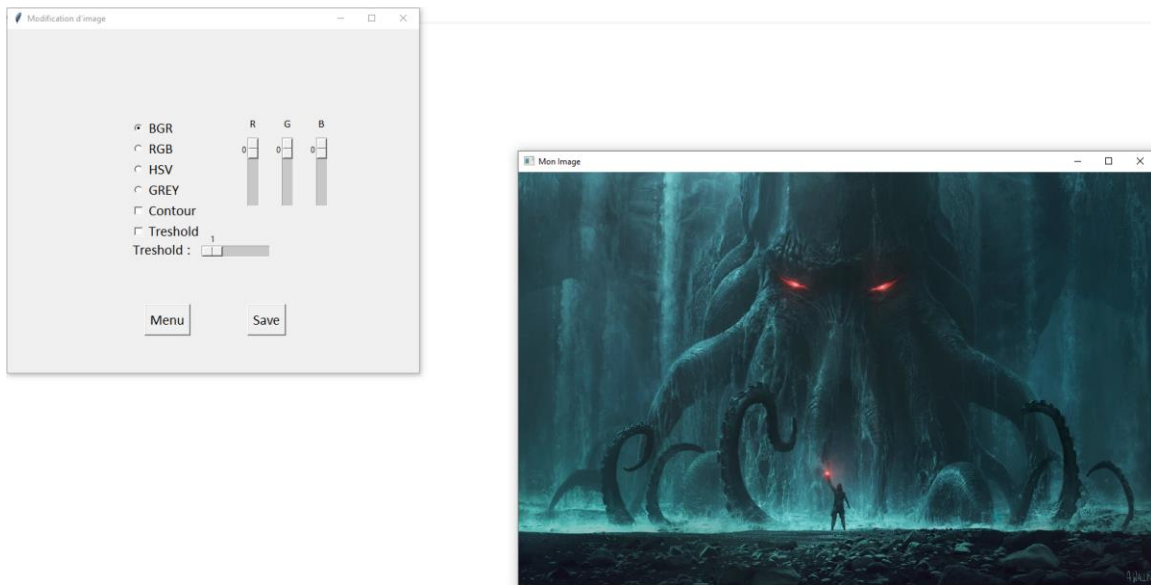


La vidéo est ensuite lue avec le mode de couleur sélectionner. L'utilisateur doit appuyer sur la touche « q », « Esc » sur le clavier ou appuyer sur le « X » dans le coin droit de la fenêtre pour fermer la vidéo.



Voici un exemple de vidéo (webcam) sous format RGB (l'image est double, car ma caméra est un casque de réalité virtuelle (- Félix)).

Pour la lecture d'image, celle-ci est d'abord convertie en format GBR, car *Matplotlib* la lit sous format RGB. Elle est ensuite affichée avec une fenêtre qui permet d'effectuer des modifications.



Les quatre « Radiobutton » permettent de choisir le mode de couleur de l'image. La case *Contour*, si elle est cochée, permet d'afficher le contour de l'image. On peut modifier les contours affichés à l'aide du « Slider » *Threshold*. Important, pour que les contours s'affichent, elles doivent avoir une couleur. Il faut donc, qu'un ou plusieurs des « slider R G B » aient une valeur différente de 0. Malheureusement, la fonction *drawContours()* n'affiche pas différente variation des couleurs (donner la valeur 100 au rouge (R) sera la même teinte que si on lui avait mit 200 à la place (il y a 8 couleurs possible et non 256^3)). Également, les contours ne sont pas détectés sur une image de format png. La case *Threshold* permet de faire afficher le *threshold* de l'image. La case *Contour* doit être décochée pour que cela fonctionne. Le « Slider » *Threshold* permet de modifier la valeur du *Threshold*. Le bouton *Menu* permet de retourner au menu et le bouton *Save* permet de sauvegarder l'image. Une boîte de dialogue va demander où enregistrer le fichier ainsi que le nom du fichier. Il est important de mettre l'extension de fichier après le nom. Les formats de sauvegarde que nous avons mis sont *jpg* et *png*. Un message de confirmation va apparaître pour dire si l'image a été enregistré ou si une erreur est survenue (Cela peut prendre quelques secondes). Seuls les contours ne peuvent pas être sauvegarder. Comme mentionner plus haut, on utilise *Matplotlib* pour sauvegarder l'image, car la fonction de *cv2* ne fonctionnait pas comme il faut. Le défaut de la fonction pour sauvegarder avec *Matplotlib*, c'est qu'elle met une bordure blanche autour de l'image.



Cthulhu Cv2
HSV.jpeg



Cthulhu CV2
RGB.jpeg



Cthulhu CV2
Threshold.jpeg



Cthulhu.jpg